

Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (currently amended) A method of error recovery of a remote method invocation (RMI) process binding processes in a network system, the method comprising:
binding a parent process with an remote method invocation RMI process; and
calling an object associated with the parent process, the object initiating a thread to perform the steps of:
monitoring the status of remote method invocation RMI processes;
determining if the RMI process is bound with the parent process, the
determining step including:
obtaining a bound uniform resource locator (URL) list from the RMI
process; and
determining whether the parent process's name is in the bound URL
list of the RMI process; and
rebinding the parent process with an active remote method invocation RMI
process when the thread determines that its parent process is not bound with an active
remote method invocation RMI process.
2. (currently amended) The method of claim 1, wherein the binding step comprises:
providing a network address of the parent process to the active remote method
invocation RMI process.
3. (currently amended) The method of claim 1, wherein the monitoring step comprises:
performing a list call to an active remote method invocation RMI process to
determine whether the parent process is bound to an active remote method invocation RMI
process.
4. (currently amended) The method of claim 3, wherein the rebinding step comprises:
performing a rebind call to an active remote method invocation RMI process.

5. (currently amended) The method of claim 1, wherein the monitoring step comprises:
calling an active ~~remote method invocation~~ RMI process to determine whether the parent process network address is registered with an active ~~remote method invocation~~ RMI process.
6. (currently amended) The method of claim 1, comprising:
binding a second parent process with a ~~remote method invocation~~ RMI process; and
calling a second object associated with the second parent process, the second object initiating a second thread to perform the steps of:
monitoring the status of ~~remote method invocation~~ RMI processes; and
rebinding the second parent process with an active ~~remote method invocation~~ RMI process when the second thread determines that the second parent process is not bound with an active ~~remote method invocation~~ RMI process.
7. (original) The method of claim 1, wherein the step of binding a parent process comprises:
binding one of an RMI daemon, a distributed task facility daemon, a log manager daemon, or a domain manager daemon, with an active RMI daemon.
8. (original) The method of claim 1, comprising:
terminating the thread when the parent process is terminated.
9. (currently amended) A network system, comprising:
a plurality of remote nodes, at least one of the remote nodes running a remote method invocation (RMI) process; and
a management server for managing the remote nodes, the management server including at least one processor for running an ~~remote method invocation~~ RMI process and at least one management process, each at least one management process being associated with an object capable of initiating a thread to perform the steps of:
~~monitoring the status of remote method invocation~~ RMI processes;
determining if the RMI process is bound with the parent process, the determining step including:

obtaining a bound uniform resource locator (URL) list from the RMI process; and

determining whether the parent process's name is in the bound URL list of the RMI process; and

rebinding the at least one management process with an active ~~remote method invocation~~ RMI process when the thread determines that the at least one management process is not bound with an active ~~remote method invocation~~ RMI process.

10. (original) The network system of claim 9, wherein the at least one management process comprises a plurality of management processes.

11. (original) The network system of claim 9, wherein the plurality of management processes comprise:

- a distributed task facility process;
- a domain manager process; and
- a log manager process.

12. (original) The network system of claim 9, wherein each of the remote nodes runs a service control manager agent process for performing server management tasks.

13. (original) The network system of claim 9, wherein the management server comprises:

- a secondary storage device, the secondary storage device comprising:
 - a data repository;
 - a depot; and
 - a web server.

14. (original) The network system of claim 9, wherein the plurality of remote nodes are arranged into at least one node group, the network system comprising a service control manager for managing the at least one node group.

15. (currently amended) A method of error recovery of a remote method invocation (RMI) process ~~binding a parent process to an remote method invocation process~~, the method comprising:

a) performing a rebind call to an ~~remote-method-invocation~~ RMI process to provide a network address and an object interface of a parent process to the ~~remote-method invocation~~ RMI process; and

b) performing an initialization call to an object associated with the parent process, the initialization call initiating a thread, the thread performing the steps of:

1) performing a list call to an active ~~remote-method-invocation~~ RMI process to determine whether the parent process is bound with the ~~remote-method invocation~~ RMI process, the list call obtains a bound uniform resource locator (URL) list from the RMI process and determines whether the parent process's name is in the bound URL list of the RMI process;

2) performing a rebind call to an active ~~remote-method-invocation~~ RMI process if the thread determines that the parent process is not bound with an active ~~remote-method invocation~~ RMI process; and

3) repeating steps 1 and 2.

16. (original) The method of claim 15, wherein the parent process is one of a remote method invocation daemon, a distributed task facility daemon, a log manager daemon, and a domain manager daemon.

17. (currently amended) The method of claim 15, wherein the step of performing a rebind call includes the step of performing a rebind call to an ~~remote-method-invocation~~ RMI daemon.

18. (currently amended) The method of claim 15, wherein the step of performing a list call includes the step of performing a list call to an ~~remote-method-invocation~~ RMI daemon.

19. (original) The method of claim 15, comprising:
terminating the thread when the parent process is terminated.